Setting Up SSH Passthrough

Category: Security & Logging In

The passthrough feature on the secure front-ends allows you to log into a system in the enclave by typing just one SSH command. The most useful way to use passthrough is with public key authentication and an SSH agent.

Once you set this feature up correctly, then each time you use SSH from your localhost to log into a NAS high-end computing system, you will be prompted for only the SecurID passcode. The SSH agent forwarding and an SSH passthrough program handle the public key authentication for you, so you will not be prompted for the passphrase of your private/public keys.

To configure passthrough using public key authentication, follow the 3 steps described in Setting Up Public Key Authentication for the SFEs. You must also copy your public key to any system in the enclave to which you want to connect using passthrough, and you need to edit the *.ssh/config* file on your localhost. Detailed information on these steps are linked below:

1. **Create SSH Public/Private Key Pair**
2. **Convert OpenSSH Key to Commercial SSH Key** (optional)
3. **Copy SSH Public Key to SFEs**
4. **Copy OpenSSH Public Key to Hosts Inside the Enclave**

   Hosts inside the enclave use OpenSSH, so you will need to copy the OpenSSH version of your public key to the hosts inside the enclave and place the key in your *.ssh/authorized_keys* file.

   Note: The permission for the *authorized_keys* file must be set to 600. Group/others write permissions on */u/username* and */u/username/.ssh* are not allowed for public key authentication.

   The following example uses lou.nas.nasa.gov as the enclave host. If you want SSH passthrough to work for other hosts inside the enclave, then repeat the steps below for each one.

   ♦ **Copy your OpenSSH public key**

   On your localhost, type:

```
your_localhost% scp ~/.ssh/id_rsa.pub
username@sfe1.nas.nasa.gov:
```

Note: *.ssh* is a directory. If it does not exist, make sure that you create a *.ssh* directory first before issuing the command below. Otherwise, it will copy the file *id_rsa.pub* to lou1 with the filename *.ssh*.

On SFE1, type:

```
sfe1% scp id_rsa.pub username@lou1:.ssh
```

- **Add your OpenSSH public key to your *.ssh/authorized_keys* file**

On your localhost, type:

```
your_localhost% ssh username@sfe1.nas.nasa.gov
```

On SFE1, type:

```
sfe1% ssh username@lou1
```

The *username* is your NAS login name.

On lou.nas.nasa.gov, type:

```
lou1% cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

```
Note: If you get the error
      /u/username/.ssh/authorized_keys: No such file or directory
after issuing the command above, likely, you have "set noclobber"
which prevents you from overwriting files. You can do "unset noclobber"
first to avoid this problem.
```

5. **Modify .ssh/config File on Your Local Host**

In your *~/.ssh/config* file on your localhost, add the entries for the hosts inside the enclave you want to access. If you do not have the *~/.ssh/config* file, create a new file called *config* in your *~/.ssh* directory and add the entries.

Template for .ssh/config

For your convenience, you can download a NAS template (a text file) for the *.ssh/config* file (attached at the end of this page). The contents of this file are also shown below. Sfe1 is used in this template. You can switch to using sfe2 if you wish to use sfe2 for SSH passthrough. Also remember to replace <NAS_login_name> with your NAS username before use.

This template should work for users who use this file only for accessing NAS systems. It this applies to you, use this template and continue with the instructions in **Step 6**.

```
Host sfe
# Replace sfe1 by sfe2 if sfe1 is unavailable
HostName                sfe1.nas.nasa.gov

Host sfe sfe?.nas.nasa.gov
Ciphers                 aes128-cbc,aes192-cbc,aes256-cbc,3des-cbc
ForwardAgent            no
MACs                    hmac-sha1

Host sfe sfe?.nas.nasa.gov dmzfs?.nas.nasa.gov sup*.nas.nasa.gov
LogLevel                info
ProxyCommand            none

Host pfe pfe-last pfe.nas.nasa.gov pfe-last.nas.nasa.gov
HostKeyAlias            pfe1.nas.nasa.gov
ProxyCommand            ssh -oCompression=no sfe /usr/local/bin/ssh-balance %h

# Add additional hosts to the list below as needed
Host *.nas.nasa.gov lou lou? cfe? pfe? bridge? sfe pfe pfe-last
ForwardAgent            yes
HostbasedAuthentication no
Protocol                2
ProxyCommand            ssh -oCompression=no sfe /usr/local/bin/ssh-proxy %h
ServerAliveInterval     10m

# Replace  with your NAS username
User                    <NAS_login_name>

# Enabling compression may improve performance for slow connections
#Compression            yes

# Uncomment this line if you are using OpenSSH 4.7 or later
#MACs                   umac-64@openssh.com,hmac-md5,hmac-sha1
```

Instructions for Creating Your Own .ssh/config

If you use your *.ssh/config* file for accessing both NAS systems and systems at other sites, you can add entries on top of the template discussed earlier. The entries take the form:

```
Host hostname
ProxyCommand ssh username@hostname.nas.nasa.gov /usr/local/bin/ssh-proxy hostname
```

Hostname is the name of the host you want to access. It can be the abbreviated hostname (such as *pfe1*) or the fully-qualified domain name (such as *pfe1.nas.nasa.gov*). Note that using bbftp requires the fully qualified domain name, thus it is a good idea to include both.

6. **Set Up SSH Agent**

Ssh-agent is a program to hold and manage the private key on your localhost and respond to key challenges from remote hosts. The private key is initially not stored in the agent and is added through the *ssh-add* program.

Ssh-agent is typically started in the beginning of an X session or a login session and you provide your passphrase to unlock your private key for this originiating session. For any outbound SSH connection to a remote host (for example, SFE1 or SFE2) made from this original session, the SSH agent remembers your private key and will respond to challenges automatically without prompting you to type in your passphrase again.

If you want to use SSH to connect from the first remote host (e.g., SFE1, SFE2) to a second remote host (e.g., pfe1) and possibly from the second remote host to a third remote host, a feature called **agent forwarding** allows a chain of SSH connections to forward all the key challenges back to the original agent, thus eliminating the need of using password or public/private keys for these connections.

In order for agent forwarding to work, your public key has to be installed already in all the remote hosts that you intend to connect to.

Instructions for UNIX or LINUX systems

- If you use csh or tcsh, to launch *ssh-agent*, type the following command

  ```
  your_localhost% eval `ssh-agent -c`
  ```

  If you use sh or bash, to launch *ssh-agent*, type the following command

  Example:

  ```
  your_localhost% eval `ssh-agent -s`
  ```

- To add your private key to ssh-agent, type the following command

  ```
  your_localhost% ssh-add private_key
  ```

  Example:

  ```
  your_localhost% ssh-add ~/.ssh/id_rsa
  Enter passphrase for /Users/username/.ssh/id_rsa: type your passphrase
  ```

Article ID: 232
Last updated: 16 May, 2011
The HEC Environment -> Security & Logging In -> Setting Up SSH Passthrough
http://www.nas.nasa.gov/hecc/support/kb/entry/232/?ajax=1